Journal of Heuristics, 5, 71–88 (1999)

New TSP Construction Heuristics and Their Relationships to the 2-Opt

HIROYUKI OKANO okanoh@jp.ibm.com
IBM Japan, Tokyo Research Laboratory, 1623-14 Shimotsuruma, Yamato, Kanagawa 242-8502, Japan

SHINJI MISONO misono@jp.ibm.com
IBM Japan, Tokyo Research Laboratory, 1623-14 Shimotsuruma, Yamato, Kanagawa 242-8502, Japan

KAZUO IWANO iwano@jp.ibm.com
IBM Japan, Tokyo Research Laboratory, 1623-14 Shimotsuruma, Yamato, Kanagawa 242-8502, Japan

Abstract. Correction heuristics for the traveling salesman problem (TSP), with the 2-Opt applied as a postprocess, are studied with respect to their tour lengths and computation times. This study analyzes the "2-Opt dependency," which indicates how the performance of the 2-Opt depends on the initial tours built by the construction heuristics. In accordance with the analysis, we devise a new construction heuristic, the *recursive-selection with long-edge preference* (RSL) method, which runs faster than the multiple-fragment method and produces a comparable tour when they are combined with the 2-Opt.

Keywords: traveling salesman problem, construction heuristic, edge-exchange heuristic, 2-Opt dependency

The traveling salesman problem (TSP) is a representative NP-hard problem, and is known to have a wide range of practical applications. Particularly for the geometric TSP, in which distance is defined in L_2 or L_∞ metric, many efficient heuristics and their implementations have been discussed. Johnson (1990), for example, extensively studied the performances of various heuristics in terms of computation time and final tour length. He dealt with construction heuristics, which build a tour from scratch, edge-exchange heuristics (local optimization), which improve a given tour, and meta-heuristics, which apply local optimization repeatedly. Bentley (1992) presented fast implementations of various construction heuristics and edge-exchange heuristics using a k-d tree, gave a data structure for fast proximity searches, and also studied the performances of the heuristics in terms of time and tour length. These studies revealed the heuristics' tradeoff between time and tour length, and underlined the practical importance of taking this tradeoff into account.

For example, when we deal with on-the-fly route optimization using an embedded computer in a drilling machine for printed-circuit boards, the available computation time and power are quite limited; thus, only a construction heuristic is used. For this purpose, Misono and Iwano (1996) devised the *grand-tour* method (GT), a variant of the *random-addition* method (RA). The GT generates a shorter tour and runs faster than the *multiple-fragment* method (abbreviated to MF, also called the *Greedy* method) which Bentley regards as one of the best heuristics.

When additional computation time and power are available, the best choices of heuristics, in increasing order of necessary computation time and decreasing order of tour length, are

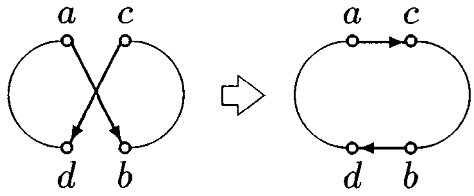

Figure 1. Edge exchange by the 2-Opt.

the 2-Opt, the 3-Opt, the variable k-Opt, and multiple-run of the variable k-Opt, where the k-Opt is an edge-exchange heuristic that deletes and inserts k edges at one time. For example, the 2-Opt swaps two edges so that the tour length decreases (Fig. 1), until no such pair of edges is left in the tour. Note that when $k \leq 3$, which means the 2-Opt or the 3-Opt is used, the edge-exchange heuristic is applied to construction tours because, as pointed out by Johnson (1990), random starting tours would greatly increase the running times for the 2-Opt and the 3-Opt, and lead to poorer final solutions.

In this paper, we consider a case in which the 2-Opt is used, and discuss what properties construction heuristics need to have when they are combined with the 2-Opt. The 2-Opt is one of the most widely used traveling salesman heuristics because, when it is used with an appropriate construction heuristic, it is the fastest way to find tours whose lengths are four to five percent in excess of the optimal values, which is practical enough for many situations. The above-mentioned GT has a better performance than the MF when neither is combined with the 2-Opt; however, we found that the final (2-Opt) tour of the GT is longer than that of the MF, which is regarded as the most appropriate construction heuristic to use with the 2-Opt (Fig. 2).

Figure 2 shows the results of various construction heuristics and how these results change after application of the 2-Opt. Results before and after the 2-Opt for each heuristic are connected by lines. (Note that the actual changes of tour lengths are not linear; they normally decrease first steeply and then gradually.) The vertical segments of the RA indicate the variances of its tour lengths. Tour lengths are shown as ratios to lower bounds $L^* = L/1.015$ estimated from the results L given by the variable k-Opt method (LK) devised by Lin and Kernighan (1973), which generates tours whose lengths are one to two percent in excess of the optimal values. The figure shows that the GT has the best performance in terms of tour length and CPU time among the *construction* heuristics examined here. When the GT is followed by the 2-Opt, however, the tours given by the GT + 2-Opt become far longer than those given by the MF + 2-Opt. (We denote a pair of a construction heuristic C and the 2-Opt by $C+$ 2-Opt, where the 2-Opt is applied to a tour obtained by C.)

More generally, use of the 2-Opt improves the tour length less for the addition heuristics—the *nearest-addition* method (NA), the *farthest-addition* method (FA), the RA, and the GT—than for the nearest-neighbor-type heuristics—the *nearest-neighbor method* (NN) and the MF. Therefore, in this paper, we address the "2-Opt dependency," and devise a construction

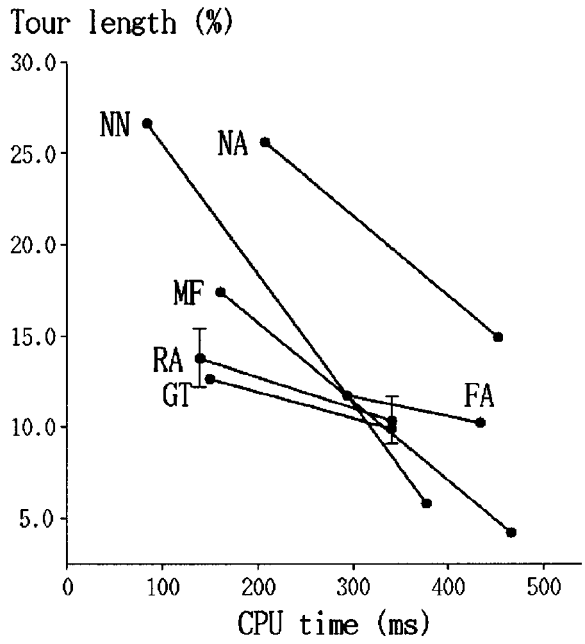

Figure 2. Performances of various construction heuristics combined with the 2-Opt.

heuristic that outperforms the MF + 2-Opt in terms of both computation time and tour length.

By the *2-Opt dependency* we mean the extent to which the *performance* of the 2-Opt depends on the initial tour created by the use of construction heuristics; here, the performance is measured in terms of execution time and tour length. The reality of the 2-Opt's dependence on construction heuristics was demonstrated by Bentley (1992) and Reinelt (1994), but no detailed analysis has yet been carried out. Perttunen (1994) examined the 2-Opt dependency more closely, but focused only on the *savings* method, a construction heuristic devised by Clarke and Wright (1964). In this paper, we analyze in detail the 2-Opt's dependence on the construction heuristics, using our own original measuring factors.

We use two factors: a *nearest-neighbor ratio* (an *NN ratio*), which measures how well a construction heuristic shapes the local structure of a tour, and an *OPT range*, which estimates the average number of edges examined by each local search of the 2-Opt. We show that the following assertions hold in most cases: (1) the higher a construction tour's NN ratio, the shorter the 2-Opt tour, and (2) the larger a construction tour's OPT range, the greater the improvement given by the 2-Opt.

In accordance with this analysis, we propose a new construction heuristic, the *recursive-selection with long-edge preference* method (RSL). In so doing, we modify the MF so that the NN ratio of the construction tour becomes higher, and at the same time the OPT range becomes larger. We show through numerical experiments both for uniformly distributed instances and for real instances that the RSL + 2-Opt runs about 15% faster than the MF + 2-Opt, and finds a comparable tour. This is the first attempt to design construction heuristics with the aim of obtaining a short tour after combination with the 2-Opt, neglecting the length of construction tours.

The paper is organized as follows: in the next two sections, the construction heuristics used in this paper are described, and the 2-Opt dependency is defined. In Section 3, two new measuring factors, the NN ratio and the OPT range, are defined. In Section 4, we describe three new construction heuristics that we devised for the experiments. One of these methods, the RSL, is the main outcome of this study. In Section 5, we analyze the local structure of a tour by using the NN ratio. In Section 6, the effects of long edges in a tour are analyzed by the use of the OPT range, and the RSL is shown to outperform the MF. Finally, Section 7 summarizes the paper.

1. Experimental Setups

We discuss the 2-Opt dependency of the grand-tour method (GT) and four other construction heuristics whose efficient implementations are described by Bentley (1992): two addition heuristics—the nearest-addition method (NA) and the farthest-addition method (FA)—and two nearest-neighbor-type heuristics—the nearest-neighbor method (NN) and the multiple-fragment method (MF). They can be stated simply as follows:

- The GT
  Creates a sequence of points $p_1, p_2, \ldots, p_n$ (an addition sequence) by using a k-d tree and the van der Corput sequence (a low-discrepancy sequence proposed by Corput (1935)) so that any subsequence $p_1, p_2, \ldots, p_k$ is uniformly distributed over the input data region, and executes the addition heuristic procedure described below with the addition sequence. See the description of the GN in Section 4 for more detail.

- The NA and the FA
  Start with a subtour consisting of a single point, and insert the nearest (or farthest) points from a subtour one by one into the places in the subtour where they least increase the tour length. The addition heuristics use a parameter in their procedures to determine the radius of their proximity search, and this parameter was set to 2.0 in our experiments (see Step 4 of the GN, defined in Section 4). We call a sequence of points inserted by these heuristics an *addition sequence*.

- The NN
  Starts at an arbitrary point and visits the nearest unvisited points one by one. Once all points have been visited, close the tour by returning to the initial point.

- The MF (the Greedy method)
  Starts with each point as a fragment of a single point, and patches the closest pairs of fragments one by one without making points of degree three or small loops.

We follow Bentley (1992) for the implementation details of the construction heuristics and the 2-Opt; we use k-d trees for proximity searches and a priority queue for selecting the closest pairs of fragments. The savings method, which Perttunen (1994) used, is not discussed here because it cannot be implemented efficiently; its naive implementation would be $O(n^2 \log n)$, where n is the number of points.

Tour lengths and CPU times are measured as the average over 100 instances of uniformly distributed points in a unit square. Tour lengths are normalized by the length of the LK times $1/1.015$. CPU times are measured on an IBM RS/6000 whose CPU is a PowerPC 112 MHz. The experiments shown below are for instances of size 1,000. We also experimented for instances of sizes up to 10,000 with the same results; they are therefore omitted in view of the space limitation.

2. The 2-Opt Dependency

As explained before, the length of a 2-Opt tour depends strongly on which construction heuristic is used for building an initial tour. This is what we mean by the 2-Opt dependency, and it is expressed by the slopes of lines in Fig. 2. For quantitative analysis of the 2-Opt dependency, we define the *slope* simply as:

$$\text{Slope} = \frac{\text{Improvement in tour length given by the 2-Opt}}{\text{CPU time of the 2-Opt}},$$

and normalize its value by the slope of the NN.

In the following sections, tours before and after application of the 2-Opt will be referred to as construction tours and 2-Opt tours, respectively. Similarly, the names of factors may be used with the prefix construction or 2-Opt, which means that the factors are observed before or after the 2-Opt, respectively.

3. The Nearest-Neighbor Ratio and the OPT Range

3.1. *The Nearest-Neighbor Ratio*

In order to measure the extent to which the 2-Opt changes the structure of a tour, we first examine the edge-length distributions. Figures 3 and 4 show the edge-length distributions and a construction tour of the MF, while Figs. 5 and 6 do so for the GT. The long edges that we see in Figs. 3 (the circled part) and 4 disappear after application of the 2-Opt, which means that long edges are shortened by the 2-Opt. However, we cannot see shifts of the distributions leftward after the 2-Opt in either Fig. 3 or Fig. 5. Thus, in order to investigate shifts of short edges in detail and examine how the 2-Opt transforms the local structure of a tour, we introduce a new factor, the *nearest-neighbor (NN) ratio*.

The NN ratio is defined as the ratio of the number of *NN edges* to the total number of edges in a tour, where an NN edge is defined as an edge at least one of whose end points is the nearest neighbor of the other end point. Note that we assume there is only one nearest neighbor for each point. For the same instance used in Figs. 3 to 6, the MF + 2-Opt's NN

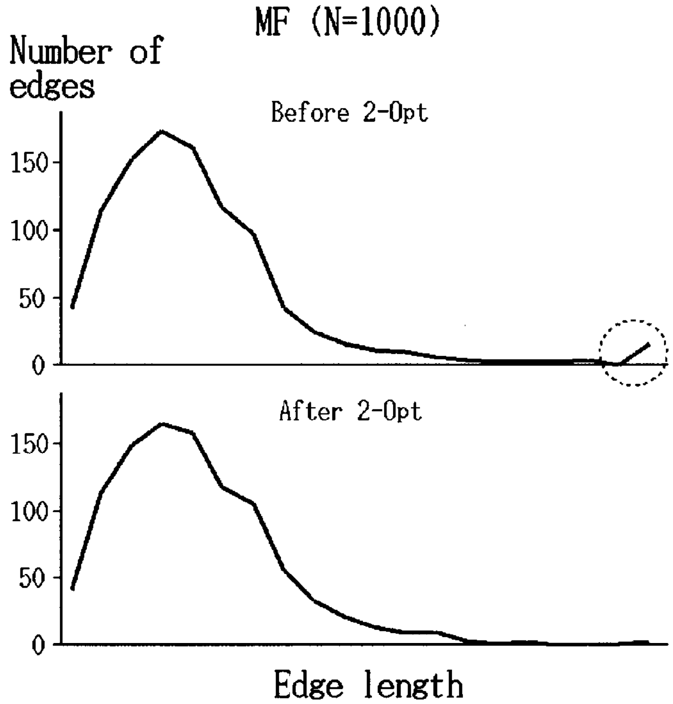

Figure 3. Edge-length distributions of the MF.

ratio is 60%, and the GT + 2-Opt's NN ratio is 54%. These values seem to reflect their 2-Opt tour lengths (Fig. 2). Note that the NN ratios of the LK tours for this instance are about 60% irrespective of which construction heuristic is used.

3.2. The OPT Range

When the 2-Opt exchanges a pair of edges, at least one of the resulting edges is shorter than either of the original pair of edges; therefore, in choosing an edge (p_i, p_j) and another edge with which to exchange it, the 2-Opt examines only edges connected to points that lie within a circle centered at p_i with radius $d(p_i, p_j)$. Because of this characteristic of the 2-Opt, long edges in construction tours expand the size of the area in which the 2-Opt can affect the structure of tours.

Figures 3 to 6 show that the construction tour of the MF contains a larger number of long edges than the GT tour (note the circled part of Fig. 3), which means the 2-Opt can perform more efficiently for the MF tours than for the GT tours. In fact, the slope of the MF is steeper than that of the GT (Fig. 2).

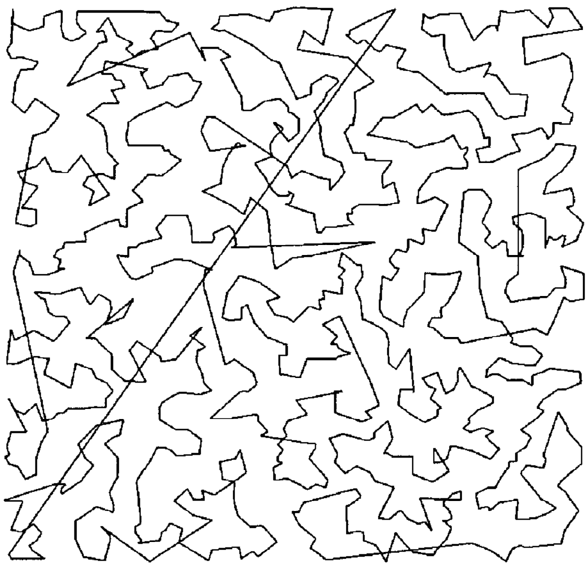

Figure 4. Construction tour of the MF.

The OPT range of a tour is defined as the average value of *cover(i)* over all points in the tour, where *cover(i)* is defined as the number of points that lie in a circle centered at a point p_i with a radius given by the edge-length $|e_i|$, excluding p_i and p_j where $e_i = (p_i, p_j)$. The average value is measured while a tour is scanned in one direction, where the tail of each edge e_i is always regarded as p_i. In the 2-Opt procedure, *cover(i)* is the number of edges that are examined as candidates with which to exchange e_i. Thus, the OPT range of a construction tour estimates the number of edges examined by each local search of the 2-Opt on the average, or the size of the area in which the 2-Opt can affect the structure of construction tours.

4. New Construction Heuristics

4.1. The Grand-Tour with NN Edge Preference (GN) Method

The GN is a modification of the GT to increase the construction NN ratio. Its procedure is as follows:

1. Build a k-d tree for input points so that they are sorted in order of tree leaves, and make an addition sequence by selecting the sorted points in order of the van der Corput sequence (a low-discrepancy sequence proposed by Corput (1935)).
2. Let the first point of the addition sequence be a subtour of a single point.

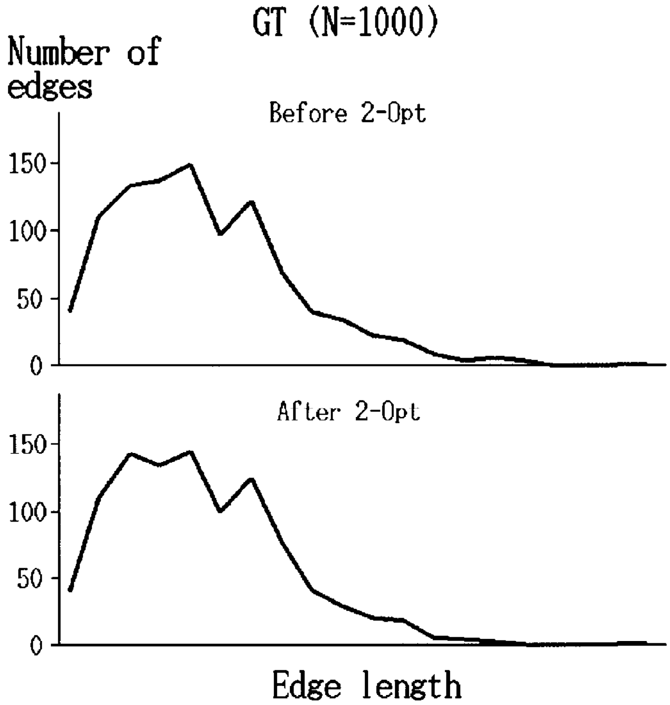

Figure 5. Edge-length distributions of the GT.

3. Select points p in the addition sequence to be inserted one by one.
4. Find all the points in a circle of radius $d(p, q) \times 2.0$ centered at p, where q is the nearest-neighbor point in the subtour from p, and choose one of the edges e connected to the points in the circle, with the following criteria:
   (a) The selection most greatly increases the number of the NN edges, or
   (b) The selection least increases the tour length.
5. Break the edge e, and insert p between the two end points of e.
6. Go to Step 3 until all the points in the addition sequence have been selected.

The only difference between the GT and the GN is criterion (a), which the GT does not have. Note that criterion (a) has priority over (b).

Note also that the van der Corput sequence is generated deterministically and is known to have the following property: let $p_1, p_2, \ldots, p_n$ be the first n points of the van der Corput sequence. Its subsequences (say, $p_1, p_2, \ldots, p_k$ for any $k \leq n$) are uniformly distributed over [0,1].

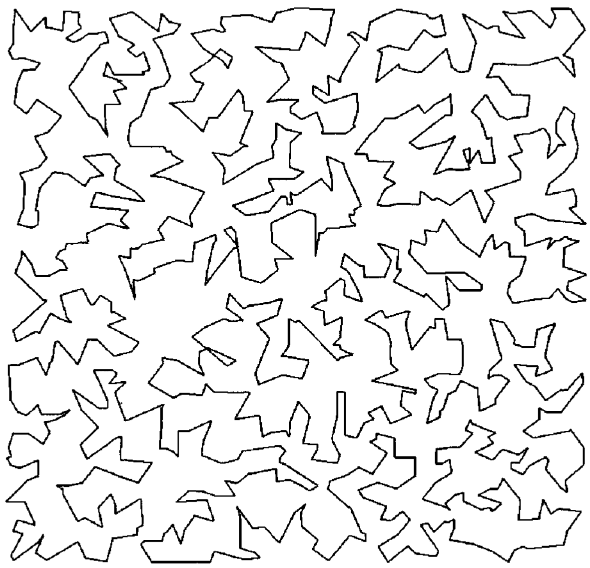

Figure 6. Construction tour of the GT.

4.2. The Recursive-Selection (RS) Method

The RS is a modification of the MF to increase the NN ratio. Its procedure is as follows:

1. Sort the input points in increasing order of the distance between each point and its nearest neighbor.

2. Select sorted points p one by one in the sorted order.

3. Connect p to its nearest neighbor, unless doing so creates points of degree three or small loops. In this case, simply cancel the selection.

4. Go to Step 2 until all the input points have been selected.

5. If any points of degree one or zero remain after all the input points have been selected, go to Step 1, assuming them to be *input points*.

If we consider a graph of all the NN edges in a tour, the degrees of nodes in the graph turn out to be fairly small for uniformly distributed instances; the maximum degree is four and the average is 1.4 for the instance used in Figs. 3 to 6. Thus, the first execution of Step 2 effectively finds nearly the maximum number of NN edges of an instance. The RS tour for the above instance includes 654 NN edges, which is only two less than the maximum.

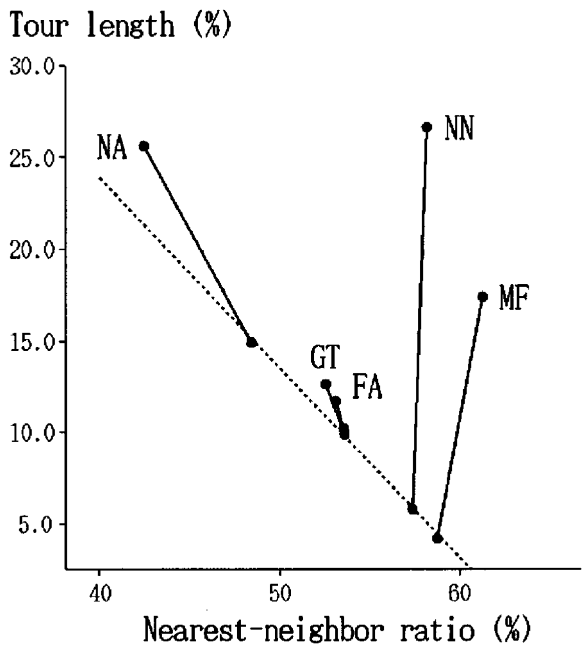

Figure 7. Changes in tour lengths and NN ratios due to application of the 2-Opt.

4.3. The RS with Long-Edge Preference (RSL) Method

To increase the OPT range, beginning from the fourth recursion of the RS procedure, change Step 2 as follows:

2. Select sorted points p one by one in *reverse* order.

This modification of the RS can be implemented simply by scanning a sort table in reverse, after the fourth recursion. Note that, here, the first execution of the procedure is regarded as the first recursion. Note also that NN edges in an RS tour are all connected in the first recursion, and that subsequent recursions do not affect the resulting NN ratio.

5. Experiments on the NN Ratio

Figure 7 shows the tour lengths and the NN ratios of both construction and 2-Opt tours (the construction and 2-Opt NN ratios) for various construction heuristics with the dotted line (the *fitting line*) that best fits the data of 2-Opt tours. The results before and after application of the 2-Opt are connected by lines. The fitting line allows us to make the observation that the higher the 2-Opt NN ratio is, the shorter the 2-Opt tour is. We can also

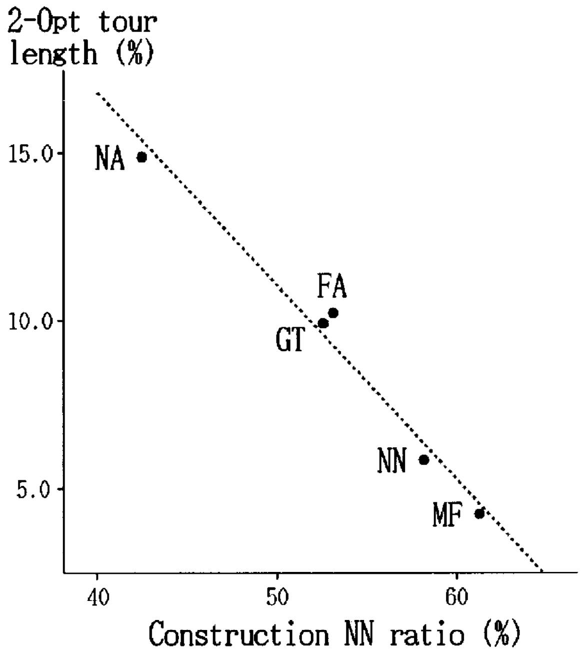

Figure 8. 2-Opt tour lengths and construction NN ratios.

see from Fig. 7 that 2-Opt NN ratios vary widely (from 48% to 59%), depending strongly on the construction NN ratios. The NN ratio of the NA method, for example, is increased from 42% to 48% by the 2-Opt, but its 2-Opt NN ratio is still much smaller than that of the MF method. In contrast, the NN ratios of LK tours concentrate around 59% and have a small variation of less than 3%. Thus, the 2-Opt, unlike the LK, does not significantly change the NN ratio. These observations suggest that there is a correlation between the construction NN ratio and the 2-Opt tour length.

In Fig. 8, we plot the 2-Opt tour lengths and the construction NN ratios of the heuristics under consideration, and add a fitting line as before. This fitting line clearly shows that the higher the *construction* NN ratio, the shorter the *2-Opt* tour. In other words, one way to obtain a 2-Opt tour of shorter length is to realize a construction tour with a higher NN ratio. This observation motivated us to devise two construction heuristics, the RS and GN methods, which are described in the previous section.

We designed the RS (resp. the GN) by modifying the MF (resp. the GT) to realize a higher construction NN ratio. By adding the construction NN ratios and the 2-Opt tour lengths of the RS and the GN to Fig. 7, we obtain Fig. 10. This shows that the construction NN ratios of the RS and the GN are higher than those of the MF and the GT by 6% (3.6 points) and 16% (8.4 points), respectively.

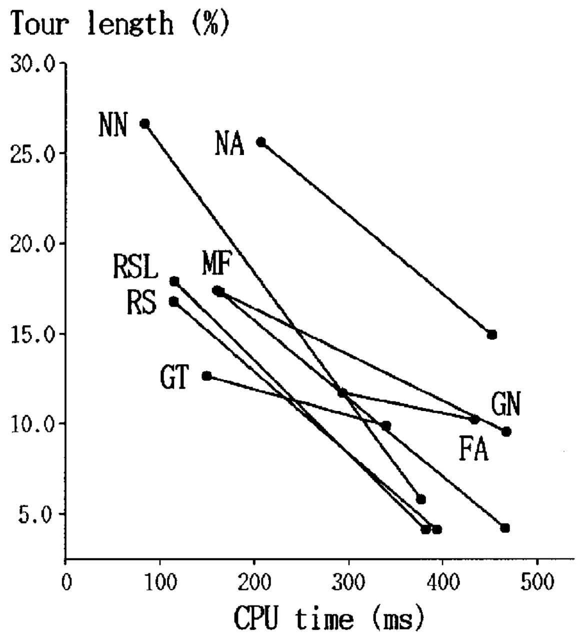

Figure 9. Performances of the RS, GN, and RSL combined with the 2-Opt.

Neither the RS nor the GN, however, improves the 2-Opt tour length significantly. The tour lengths of the RS + 2-Opt and the GN + 2-Opt are shorter than those of the MF + 2-Opt and the GT + 2-Opt by 0.1% and 0.3%, respectively. Increasing the construction NN ratio, on the other hand, has little influence on the slope (Fig. 9). Furthermore, we can see a gap in the 2-Opt tour length between two types of heuristics—nearest-neighbor (RS, MF, and NN) and addition (GN, GT, NA, and FA)—irrespective of the construction NN ratio (Fig. 10). The NN ratio, thus, is not a definite factor for characterizing the 2-Opt dependency; rather, it is a valid factor (for each type of heuristic) for measuring a tour's quality.

Although the construction NN ratio does not clarify the 2-Opt dependency, it allows us to find heuristics for better performance. The RS method, when combined with the 2-Opt, runs 15.6% faster than the MF, and its 2-Opt tour length is comparable to that of the MF (Fig. 9). This is because the number of nearest-neighbor searches performed by the RS is only 66.4% of the number performed by the MF, and also because the RS does not require insert/delete operations on a heap as the MF does, and only scans the sorted list. The number of input points for each recursion of the RS, which is equal to the number of nearest-neighbor searches, for example, changes as follows: $1000, 681, 189, 52, 14, 4$.

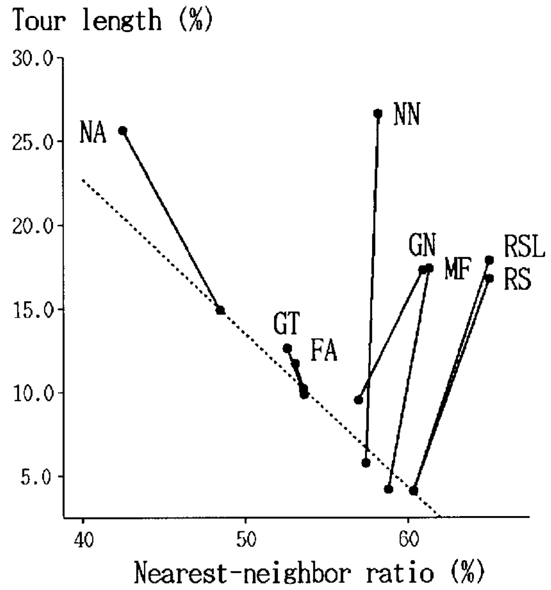

Figure 10. Changes in the tour lengths and NN ratios of the RS, GN, and RSL due to application of the 2-Opt.

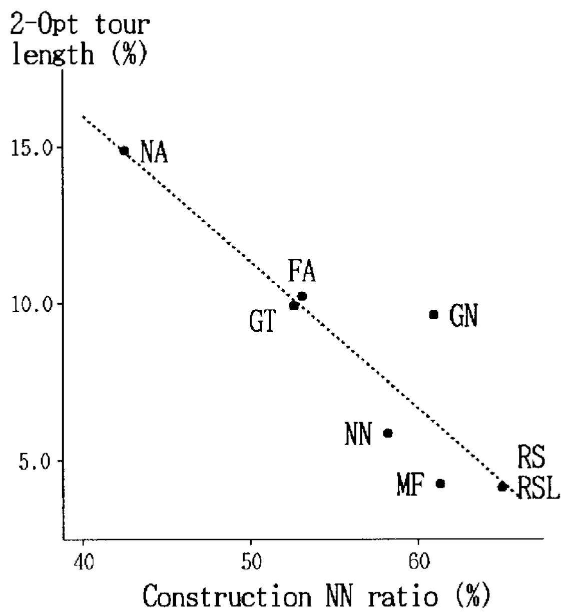

Figure 11. 2-Opt tour lengths and construction NN ratios of the RS, GN, and RSL.

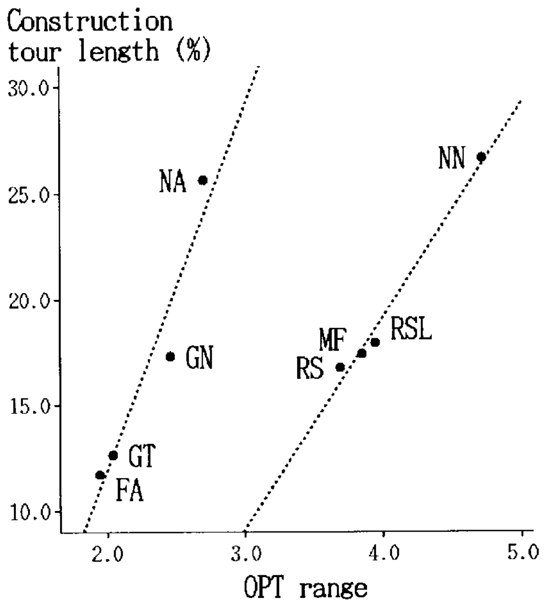

Figure 12. Construction tours and OPT ranges of the RS, GN, and RSL.

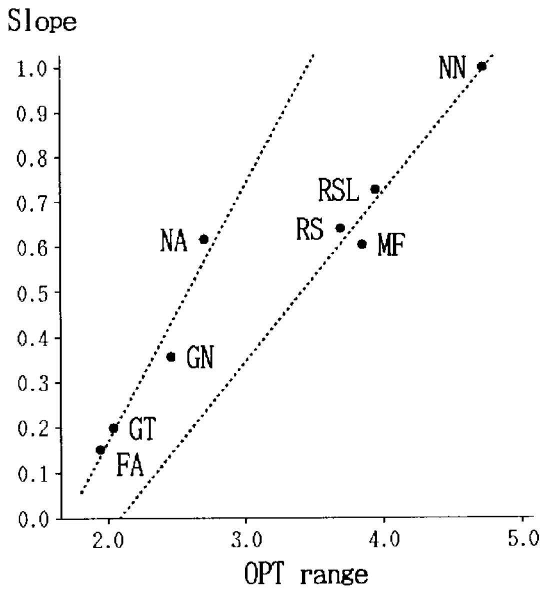

Figure 13. Slopes and OPT ranges of the RS, GN, and RSL.

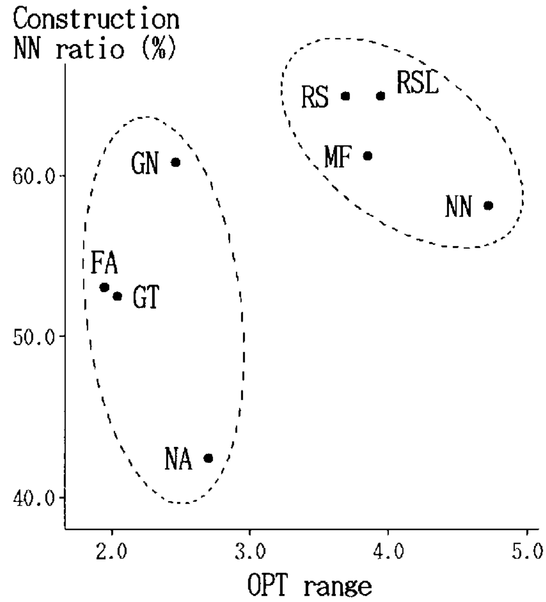

Figure 14. Construction NN ratios and OPT ranges of the RS, GN, and RSL.

6. Experiments on the OPT Range

As explained in Section 3, the OPT range of a construction tour estimates the size of the area in which the 2-Opt can affect the structure of the construction tour. The OPT range also measures the effect of long edges in a construction tour on the performance of the 2-Opt. In this section, we observe the OPT ranges of various tours, and show that the OPT range can measure the 2-Opt dependency.

Figure 12 shows the construction tours and the OPT ranges of various construction heuristics with fitting lines; here, the fitting lines are drawn separately for the nearest-neighbor-type heuristics and for the addition heuristics. The fitting lines show that the two types of heuristics form different groups, however, they have the same trend: the smaller the OPT ranges, the shorter the construction tours. This is easily understood, because the OPT range reflects the average edge-length. Figure 13 shows the slopes and the OPT ranges of various construction heuristics with fitting lines. We can see almost the same trend as in Fig. 12, which means that the larger the OPT ranges, the greater the slopes; that is, the greater the improvement that the 2-Opt gives. Therefore, we can use the OPT range of a construction tour as an estimator of the 2-Opt dependency.

Our next question is how to increase the slopes without affecting the lengths of the construction tours. Figures 12 and 13 show that the two objectives, decreasing the length of the construction tours and increasing the slopes, require a tradeoff. We must, therefore, seek a better tradeoff in order to improve the construction heuristics, instead of just changing the

Table 1. Application to TSPLIB instances (Part I).

(Tour lengths are expressed as percentages in excess of the optimal values)

Instance	MF + 2-Opt tour length	RSL + 2-Opt tour length	Total CPU time $\left(\frac{RSL + 2\text{-}Opt}{MF + 2\text{-}Opt}\right)$	2-Opt CPU time $\left(\frac{2\text{-}Opt\ after\ RSL}{2\text{-}Opt\ after\ MF}\right)$
a280	5.8%	4.6%	88.3%	91.0%
berlin52	10.4	6.6	98.0	129.9
bier127	4.9	5.1	87.5	105.0
ch130	6.8	1.4	59.6	58.2
ch150	3.8	1.9	69.0	71.7
d198	3.8	2.9	87.0	96.1
d493	3.5	4.1	83.4	90.5
d657	4.2	4.7	86.2	95.1
d1291	6.9	6.7	89.2	92.8
d1655	4.8	4.7	87.9	90.6
dsj1000	6.4	4.8	89.4	96.8
fl417	3.8	4.0	97.2	109.6
fl1400	2.4	4.1	82.9	84.9
fnl4461	3.5	3.4	78.6	82.4
lin105	2.6	0.7	78.6	96.4
lin318	3.9	3.4	98.3	131.3
nrw1379	4.0	3.6	60.7	59.7
p654	1.3	3.2	80.2	80.7
pcb442	4.1	3.9	83.7	86.3
pcb1173	4.9	4.2	86.7	94.9
pcb3038	4.5	4.2	81.6	86.7
pla7397	4.5	5.1	96.0	98.3
pr76	3.4	4.6	75.5	78.1
pr107	0.5	2.4	101.4	114.3
pr124	1.0	1.1	98.4	136.2

slopes. The OPT range and the construction NN ratio, on the other hand, do not involve a tradeoff (Fig. 14), because the OPT range is mainly increased by long edges that are not NN edges. In Fig. 14, we cannot see any clear correlation between the OPT range and the construction NN ratio; the direction of a good (tour/slope) tradeoff in this figure is toward the upper right. Figure 14 also shows that the group of nearest-neighbor-type heuristics, enclosed by a dotted circle, has a better tradeoff than that of addition heuristics.

In devising the RSL method, we modified the RS in such a way as to increase the OPT range and keep the NN ratio high. Figure 9 shows that the slope of the RSL + 2-Opt is steeper than that of the MF + 2-Opt. It also shows that the 2-Opt tour of the RSL is comparable to that of the MF, while the construction tour of the RSL is longer than that of the MF. Note that the 2-Opt for the RSL can be computed faster than the 2-Opt for the MF or the RS, and that the RSL + 2-Opt shows a consequent improvement over the MF + 2-Opt of 18.0%, which is larger than that of the RS + 2-Opt (15.6%). The RSL's improvement over the MF results from the differences in the numbers of nearest-neighbor searches they require and in their computation times for a quick-sort and a priority queue.

Figure 11 shows that the construction NN ratio of the RSL is exactly the same as that of the

Table 1. Application to TSPLIB instances (Part II).

Instance	MF + 2-Opt tour length	RSL + 2-Opt tour length	Total CPU time $\left(\frac{RSL + 2\text{-}Opt}{MF + 2\text{-}Opt}\right)$	2-Opt CPU time $\left(\frac{2\text{-}Opt\ after\ RSL}{2\text{-}Opt\ after\ MF}\right)$
pr136	9.9%	9.6%	81.7%	90.2%
pr144	1.2	3.5	82.2	91.1
pr152	1.6	3.1	109.3	150.8
pr226	2.6	3.3	86.7	93.6
pr264	5.6	4.5	92.6	92.5
pr299	5.9	3.9	86.1	94.9
pr439	5.3	6.5	88.6	97.4
pr1002	5.4	5.1	79.4	85.3
pr2392	5.6	4.9	89.3	94.7
rl1304	4.3	3.3	83.1	91.3
rl1323	3.8	3.3	75.4	79.3
rl1889	5.4	4.8	77.5	81.8
ts225	2.9	3.0	99.4	103.0
tsp225	1.7	0.9	78.2	86.9
u159	7.2	5.0	66.9	66.0
u574	4.4	4.3	83.5	91.7
u724	3.8	3.7	77.4	82.9
u1060	4.6	3.5	80.5	85.9
u1432	5.2	5.5	91.3	92.3
u1817	4.2	4.9	90.9	94.1
u2152	5.8	5.8	96.7	101.3
u2319	2.5	3.3	93.9	94.1
usa13509	4.3	4.0	46.6	45.3
vm1084	5.6	4.6	85.0	94.0
vm1748	5.1	4.8	88.0	95.6
Average	4.4%	4.1%	84.7%	92.7%

RS. Figures 12 and 13 show that the RSL succeeds in increasing the slope without greatly increasing the construction tour length (see also Fig. 9). Figure 14 shows that the RSL has a better tradeoff between the tour and the slope than the other construction heuristics we tested. Note that the direction of a good tradeoff is toward the upper right in Fig. 14.

Table 1 shows that the RSL + 2-Opt's advantage over the MF + 2-Opt is the same for real instances from TSPLIB collected by Reinelt (1991). In the table, tour lengths given by the MF + 2-Opt and the RSL + 2-Opt as percentages in excess of the optimal values are shown, and CPU times used by the RSL + 2-Opt as percentages of those used by the MF + 2-Opt are also shown. The RSL + 2-Opt is faster than the MF + 2-Opt by 15.3% on the average, and generates tours whose length are 4.1% in excess of the optimal values.

7. Conclusion

We have analyzed how and why the performance of the 2-Opt depends on the initial tour built by construction heuristics, and studied how to modify a construction heuristic to obtain a better performance when the 2-Opt is applied after it. For the analysis, we defined the slope as a quantitative evaluation of the 2-Opt dependency, and introduced two new measuring factors, the NN ratio and the OPT range. We showed that the NN ratio measures a tour's quality, and that increasing the NN ratio of a construction tour improves the 2-Opt tour without affecting the slope greatly. We also showed that the OPT range measures the 2-Opt dependency of a construction tour, and that increasing the OPT range of a construction tour makes the slope steeper.

On the basis of the analysis with these factors, we devised a new construction heuristic, the *recursive-selection with long-edge preference* (RSL) method, which is about 15% faster than the multiple-fragment (MF) method, and finds a comparable tour to the MF method when the 2-Opt is used for postprocessing. This is one of the first attempts to analyze construction heuristics in conjunction with the improvement heuristic, and the first to design construction heuristics so that they perform well when combined with the 2-Opt.

References

Johnson, D. S. (1990). "Local Optimization and the Traveling Salesman Problem." In *Proceedings of the Seventeenth Colloqium on Automata, Languages and Programming*, pp. 446–461. New York: Springer.

Bentley, J. L. (1992). "Fast Algorithms for Geometric Traveling Salesman Problems," *ORSA Journal on Computing* 4(4), 387–411.

Lin, S., and B. W. Kernighan. (1973). "An Effective Heuristic Algorithm for the Traveling-Salesman Problem," *Operations Research* 21, 498–516.

Misono, S., and K. Iwano. (1996). "Experiments on TSP Real Instances," *IBM Research Report*, RT0153.

Reinelt, G. (1994). *The Traveling Salesman, Computational Solutions for TSP Applications*. Heidelberg: Springer.

Perttunen, J. (1994). "On the Significance of the Initial Solution in Traveling Salesman Heuristics," *Journal of the Operational Research Society* 45(10), 1131–1140.

Clarke, G., and J. W. Wright. (1964). "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points," *Operations Research* 12, 568–581.

van der Corput, J. G. (1935). "Verteilungstunktionen I, II," *Nederal. Akad. Wetensh. Proc. Ser. B* 38, 813–821, 1058–1066.

Reinelt, G. (1991). "TSPLIB—A Traveling Salesman Problem Library," *ORSA Journal on Computing* 3, 376–384.